**整数**

-eq 相等

-ne 不等

-gt 大于

-lt 小于

-ge 大于等于

-le 小于等于

**文件权限**

-r 是否可读

-w 是否可写

-x 是否可执行

**文件类型**

-e (文件或目录)是否存在

-f 存在且为文件

-d 存在且为目录

**两个文件比较**

file1 -nt file2 检查file1是否比file2新

file1 -ot file2 检查file1是否比file2旧

**字符串(双引号””不能省略)**

-n 是否为非空字符串

-z 是否为空字符串(字符串长度为0)

“Strging1” = “String2” 字符串String1和String2是否相等

“Strging1” != “String2” 字符串String1和String2是否不相等

## []

**边界必须为空格**

支持布尔逻辑(&&、||)来组合测试,如下所示:

[root@localhost ~]# [ -d sed.txt ] || [ -f sed.txt ] # 逻辑或

[root@localhost ~]# echo $?

0

[root@localhost ~]# [ -d sed.txt ] && [ -f sed.txt ] # 逻辑且

[root@localhost ~]# echo $?

1

## 双括号(())

边界不需要为空格

双括号命令的格式如下：

(( expression ))

expression可以是任意的数学赋值或比较表达式

|  |  |
| --- | --- |
| **符号** | **描述** |
| val++ | 后增 |
| val-- | 后减 |
| ++val | 先增 |
| --val | 后减 |
| ! | 逻辑求反 |
| ~ | 位求反 |
| \*\* | 幂运算 |
| << | 左位移 |
| >> | 右位移 |
| & | 位布尔且 |
| | | 位布尔或 |
| && | 逻辑且 |
| || | 逻辑或 |

[root@localhost ~]# ((4 \*\* 3 > 34 || 4 \*\* 3 < 43))

[root@localhost ~]# echo $?

0

[root@localhost ~]# ((4 \*\* 3 > 34 && 4 \*\* 3 < 43))

[root@localhost ~]# echo $?

1